09851142 . 050901

APPLICATION FOR UNITED STATES LETTERS PATENT

by

ROBERT C. NEWBOLD

for

MULTIPLE PROJECT SCHEDULING SYSTEM

Burns, Doane, Swecker & Mathis, LLP
Post Office Box 1404
Alexandria, Virginia 22313-1404
(703) 836-6620

Attorney Docket No. 033182-001

MULTIPLE PROJECT SCHEDULING SYSTEM

Field of the Invention

[0001] The present invention is directed to a computerized system for scheduling multiple projects, and more particularly, to a multiple-project, resource-constrained project scheduling system and a method for determining the earliest feasible time that a project should be in initiated or introduced into an organization.

Background of the Invention

[0002] Project management and resource scheduling are important functions that can be applied to a variety of enterprises and business concerns. There are many variables and dependencies that are involved in a project management effort. These variables and dependencies must be managed and resolved in an organized and effective manner to achieve desired and optimal results.

[0003] In general, there are three main elements that should be addressed in scheduling a project. First, the project requirements are determined and quantified. These requirements, for example, can include resource availability, manufacturing schedules, engineering milestones and other tasks that must be performed to achieve an end result. These project requirements are then represented in a model or framework to organize and track all of the requirements. There are many types of models that can be used to represent and manipulate these project requirements. For example, these models can take into account factors such as tasks, task interrelationships, precedence relationships, resources and various restrictions on the tasks and resources that make up a project.

[0004] The second main element in scheduling a project is an objective function. This objective function determines the relative quality and comparative advantages and disadvantages between various schedule alternatives at different decision points. The objective function can determine which schedule is a "better" schedule and which schedule is a "worse" schedule. Often included in this

objective function is what-if analysis to evaluate amongst an array of possible outcomes and scenarios and provide decision support to management and other interested members of an organization. As the name suggests, the objective function expresses the "objectives" of the schedule, usually in mathematical terms and quantities that can be more easily managed and compared to each other.

[0005] The third main element in scheduling a project is the actual process for determining task sequences and times. Many processes exist in the prior art for scheduling one or more projects. These processes can support various kinds of modeling capabilities and objective functions. Some of these are heuristically based and some are designed to produce optimal results. Objective functions normally focus on bottom-line results predicted by a schedule. These bottom-line results are typically based on due dates, predicted completion dates, milestone completions and resource productivity. Optimization techniques may also be applied to quantities such as weighted values, a sum or a maximum value in order to arrive at an optimal solution. Another technique used is phase-gate processes. Phase-gate processes establish different project phases. Each phase ends with a gate that may or may not be passed through, depending upon the satisfaction of specific entry or exit criteria.

[0006] There are limitations associated with many of these existing techniques. Although the theories underlying these techniques may be sound, in a practical implementation it is often very difficult, if not impossible, to produce a truly optimal schedule. There are numerous reasons for this difficulty. A schedule is at best an imperfect model or representative approximation of reality. Reality is a constantly changing and evolving entity, which often requires constant attention and changes to schedules and their management, further complicating the modeling activity. Furthermore, most practical scheduling problems are "NP-hard",

10060050.024102

i.e., nondeterministic polynomial, meaning that they are most likely very difficult to solve optimally using a mathematical analysis.

[0007] Another complication in project scheduling is that it is quite common for organizations is to have more active projects in place than their resources can easily or effectively accommodate. This can quickly turn into an overload condition where there are insufficient resources available at the necessary or opportune time to have the intended effect. One reason that this problem can occur is that management wants to take advantage of most, if not all, of the new opportunities that currently exist or will exist shortly. Problems can also arise when front-end organizations, like marketing, may have sufficient capacity and resources to propose and define projects. At the same time, however, other organizations, such as new product development and manufacturing, may be currently operating at capacity and/or over-subscribed with work, and cannot handle the new work that marketing is committing them to do. Further in this regard, Little's Law says that work in a system or work-in-process (WIP) is proportional to flow time. In other words, the more work that exists in the organization at a given point in time, the longer the average start-to-finish time for any of the given projects.

09851426.050601

[0008] There are other, less obvious penalties to having too many active projects occurring simultaneously in an organization. Probably the most significant penalty is the one attributed to multitasking inefficiencies. This occurs when personnel move back and forth between tasks and/or projects without completing any of them. The end result of these multitasking inefficiencies is that the time required to complete any given task can be extended arbitrarily. In a competitive environment where time to market is critical, this arbitrary task extension can be a significant liability and put an organization at a severe competitive disadvantage in the marketplace.

[0009] Finally, it is almost a universal truth that all projects will encounter some type of delay, both foreseen and unforeseen. These delays can arise from known or unknown origins and can extend projects significantly.

[0010] Once launched, projects progress in ways that may not have been envisioned or planned for in the initial project stages. There are known and unknown events that can introduce all manners of perturbations into the system with correspondingly different results and outcomes. Projects can take on a life of their own, and it becomes difficult to make a decision to terminate them before completion. This usually happens only after significant time, energy and resources have been expended. The termination of a project can be a very stressful, wasteful and time-consuming process. There is a strong need for better control and insight into how and why projects are launched or introduced into an organization to begin with.

[0011] There are various quantitative methods that can be used to limit the introduction of new projects into organizations. One method compares the estimates of aggregate resource consumption versus availability, over a given time period. In certain simpler and less complex applications, this method can be very effective. However, there can be surprises and unforeseen results due to its approximate nature and dependency upon estimates.

[0012] Another approach is to create detailed schedules for all of the projects. Resource contention is resolved within and between the various project schedules, at least for some of the resources. The objective function attempts to optimize a combination of parameters in order to complete projects as quickly and efficiently as possible. This approach can be reliable, given reasonable project schedules. However, there is a common, practical problem encountered when scheduling projects: how much earlier should a project be started, in order to produce some minimal additional efficiency? For example, it might be possible to begin work on

10050.01420514960

a project a month earlier and, by taking advantage of temporary availability of some resources, thereby advance its completion date by a week. However, this approach may create other disadvantages, due to the fact that there are more concurrent projects in the system. A scheduling system should enable a manager to evaluate the various tradeoffs associated with such an approach.

[0013] Therefore, it is desirable to provide a practical and controllable mechanism for limiting the starts of new projects and a method that allows organizations to deal with both capacity issues and work-in-process issues. Given the common real world limitations and operational scenarios that are frequently encountered, such as project opportunities that can arrive at arbitrary times and projects that are hard to control and hard to stop, a very effective way of controlling work-in-process is to control the induction or commencement of new projects in an organization. The present invention achieves this objective through a practical, quantitative means of controlling project starts based on a combination of capacity and project flow times.

Summary of the Invention

[0014] It is an objective of the present invention to provide the user with a tool to resolve conflicts due to path dependencies, resource contention and other restrictions that may exist within individual projects. The present invention is based upon the consideration of the flow time for a project. By providing a flexible means of adjusting the flow time, the management of multiple tasks is facilitated by allowing the work in process to be maintained at acceptable levels. The method can identify resource contention issues for user-selectable resources between projects, and thereby assist the user in the appropriate course of action to take to resolve such issues. Once the issue has been resolved, the invention is able to determine where a project load should be placed in time.

10050141 - 035301

[0015] The present invention can also be used to pace or phase in the starts of one or more projects selected by the system user to be introduced into an organization. This enables the user to schedule or reschedule any number of projects using the results of the invention combined with prior status and current update information.

[0016] These and other features of the present invention are explained in greater detail hereinafter, with reference to exemplary embodiments of the present invention illustrated in the accompanying drawings.

Brief Description of the Drawings

[0017] Figure 1 is a time line illustrating multiple projects that are used as example to explain the invention;

[0018] Figure 2 illustrates an initial resource loading condition for the given set of projects depicted in Figure 1;

[0019] Figure 3 illustrates a project diagram showing the temporal relation between three tasks for a new project to be scheduled;

[0020] Figure 4 illustrates the overall process flow for the present invention;

[0021] Figure 5 illustrates an exemplary first iteration of step 2 shown in Figure 4; and

[0022] Figure 6 illustrates an exemplary secondary iteration of the step 2 shown in Figure 4.

Detailed Description

[0023] The present invention is directed to environments in which multiple concurrent projects can be ongoing at any time, and new projects need to be scheduled as they arise. The projects could be of any of a variety of different types, such as manufacturing jobs, engineering programs, design schedules, etc.

A general overview of multiple projects, to which the principles of the present invention are applicable, is depicted in Figure 1. This figure illustrates four concurrent projects, each of which comprises multiple tasks. A "task" is any activity that is to be scheduled for a to-be-determined time period. Besides indicating an actual activity, tasks can also represent placeholder or buffer times during which no activity is taking place. In the illustrated example, each task utilizes one of a number of shared resources, denoted by the blocks A, B, C, and D. Each resource could be representative of any one of a variety of different components in the completion of the project, such as a machine that is required to process materials or information, a person with a specialized skill, storage space for raw materials, a testing or quality control laboratory, etc. In the discussion which follows, it is assumed that only one unit (machine, person, etc.) is available for each of the individual resources A, B, C and D. It will be appreciated, however, that any resource could have available more or less than one of the associated units.

09851442.050801

[0024] The horizontal dimension shown in Figure 1 is time, and the length of each block represents the amount of time that each resource will be dedicated to a given task. In this example, the time units shown are arbitrary. Those skilled in the art will appreciate that these time increments can be any time increment such as hours, days, weeks, months, etc. In the illustration of Figure 1, the resource loading for each project is depicted in seriatim, i.e., as if each resource was available as soon as it was needed for the project. In reality, however, each resource can only be employed to work on one task at any given instant. Thus, for example, Project 3 cannot utilize Resource C until it has been released by Project 2. Hence, one issue that should be resolved in any project scheduling system is contention among the various projects for a set R of shared resources. In

the present case, Project 3 can be put on hold until such time as the first task of Project 2 is completed and Resource C becomes available.

[0025] Figure 2 is a time line which represents the multi-project example of Figure 1 after the individual resources A-D have been allocated among the projects in such a way that they satisfy both the project requirements (e.g., in Project 1, the task requiring Resource A must follow the task utilizing Resource B) and resource availability. The set of resources can be represented mathematically by R = {A, B, C, D}. In the example, there is one unit available for each resource contained in R. The blocks indicated for each resource represent the times that resources are unavailable due to the scheduling of Projects 1-4. Resource A is unavailable for the time increments shown in blocks A1 and A2. Resource B is unavailable for the time increments shown in blocks B1 and B2. Resource C is unavailable for the time increments shown in blocks C1 and C2. Resource D is unavailable for the time increments shown in block D1. Each block might represent actual time for which the resource has been scheduled. Alternatively, a portion of a block, or even an entire block, might be used for buffer time to reserve protective capacity for unscheduled work items that may arise unexpectedly.

09851142.050901

[0026] The information depicted in Figures 1 and 2, i.e. the tasks associated with each project, the amount of time and resources allocated to each task, and the loading of each resource, can be stored in a suitable database (not shown). In addition, this information can be graphically represented to a user on a suitable display, for instance in the manner illustrated in the figures.

[0027] As an illustrative example which facilitates an understanding of the present invention, the starting time for a new project as shown in Figure 3 will be determined. This project consists of two tasks 302 and 304 that each require a single time unit of resource C, and a task 306 requiring two time units of resource B. The arrows between the blocks represent precedence relationships. In this

particular case, the second task 304 must start after the first task 302 is completed. The task 306 must start after the second 304 task is completed.

[0028] The present invention is based upon consideration of the flow time for a project. In many situations, control over the flow time is significant to the management of multiple tasks since it enables work in process to be maintained at acceptable levels, which in turn facilitates efficient project execution. The following terms and definitions are presented for understanding the present invention prior to the detailed description that follows:

$EPST_i$ = earliest possible start time for project i.

FE_i = the maximum allowable flow expansion value for project i. The flow expansion value is ≥ 1.0.

$R = \{r_j\}$ = a set of resources for which inter-project contention is being resolved.

FT_i (t,x) = the calculated flow time of project i. It is the start-to-finish duration of the project, typically represented by the difference between completion of the latest task and initiation of the earliest task [(latest task finish) - (earliest task start)] when tasks for project i are scheduled no earlier than time t, taking into account any desired intra-project placement restrictions (including, for example, path dependencies, task durations, resources and task starting or finishing restrictions). Additional factors can be taken into account as well, such as a buffer period after the completion of the last task for the project. When **x = true,** the loading requirements of projects 1 through i-1 for R are considered when scheduling tasks for project i. As far as possible, project i tasks should not be permitted to cause additional overloads for resources in R. When **x = false,** loading for R caused by projects 1 through i-1 is not considered.

PS_i (t,x) = the start time at which the earliest task of project i would be scheduled, with t and x as defined above.

CST = current starting time for project i

ST_i = calculated start time for project i

δ = the minimum time to move CST for each iteration of the scheduling process of the present invention.

[0029] In accordance with the present invention, the maximum allowable flow expansion FE_i is a user-determined value that provides a quantitative, and hence readily perceptible, factor for controlling the scheduling of projects. Figure 4 is flow chart that illustrates the process for determining the appropriate start time for the project shown in Figure 3. In order to calculate the start time for project i, step 1 sets the current start time (CST) equal to the earliest permissible start time for project i ($EPST_i$). This serves as an initialization step. Next, in step 2 of Figure 4 there is a decision that can result in one of two outcomes. The test condition shown in step 2 is: whether the flow time of project i, scheduled to start at time CST and taking into account inter-project resource contention on the selected resources R (i.e., x=true), is less than or equal to the flow expansion coefficient FE_i times the flow time of project i scheduled to start at time CST, without taking into account inter-project resource contention (i.e, x=false).

[0030] If the calculation of step 2 results in an answer of NO, this is an indication that the flow time for the project initiated at time CST is too long. Subsequently, the process moves to step 3 of Figure 4. In step 3, CST is first set equal to the earliest scheduled task start time for project i, using the function PS_i (CST, True). This assignment can be omitted, but it helps to reduce overall processing time when it is employed. CST must then be increased by either the value for δ or the differences of the parameters calculated in step 2, i.e. $FT_i(CST,True)-[FT_i(CST,False)\times FE_i]$, whichever value is larger. Then the flow goes back to step 2 for one or more further iterations.

[0031] However, if the result of step 2 results in an answer of YES, then the process moves on to step 4. The start time for project i is set to equal the earliest scheduled task start time for project i, using the function PS_i (CST, True).

[0032] For the purposes of the illustrative example depicted in Figures 1-3, assume the following initial starting conditions for determining the starting time for the new project:

[0033] 1. Assume that $EPST_i$ is equal to zero. This means that project i can start as early as time 0.

[0034] 2. The flow expansion coefficient $FE_i = 1.5$. This means that the project's start-to-finish duration will be allowed to expand by up to 50% in order to place the project earlier.

[0035] 3. The minimum increment to increase CST is 1 time unit, where $\delta = 1$.

[0036] In step 1 of Figure 4, CST is set to $EPST_i = 0$ as an initial proposed starting time. Then for step 2 of Figure 4, loading is calculated for resources R with the addition of project i, in order to calculate FT_i(CST, True).

[0037] The resulting resource loading condition is shown in Figure 5. The loading added by the three tasks of project i (302, 304, 306) is shown by the cross-hatched areas of Figure 5.

[0038] In Figure 5, the flow time (501) for project i, taking into account the R resources, equals the time between the start of the first C task (302) and the end of the B task (306). This means that FT_i (CST, True) = 10 time units. Without taking into account the resources of R, project i's flow time would be FT_i (CST, False) = 4 time units, i.e. the sum of only the shaded areas in Figure 5. Therefore, the maximum allowable flow time is FT_i (CST, False) x FE_i = 4 x 1.5 = 6 units. Since this value is less than Ft_i (CST, True) = 10, the result of the test

in step 2 of Figure 4 is not true or NO. Therefore, the process continues to step 3 of Figure 4.

[0039] From the above calculations, FT_i (CST, True) - [FT_i (CST, False) x FE_i] = 4. This is larger than the value of δ. Therefore in step 3 of Figure 4, CST is incremented by a value of 4 units to establish a new proposed starting time. The process then returns to step 2 for another iteration through the process, the results of which are shown in Figure 6.

[0040] Figure 6 shows the loading across R during calculations for the second iteration of step 2. The resulting flow time FT_i (CST, True) is 12 - 6 =6. Since FT_i (CST, False) is still 4, the maximum allowable flow time FT_i (CST, False) x FE_i = 4 x 1.5=6 is equal to the calculated flow time FT_i (CST, True) = 6. Therefore, the test result from step 2 of Figure 4 is true or YES, and the process proceeds to step 4 of Figure 4.

[0041] Based on Figure 6, it can be seen that PS_i (CST, True) equals 6. Therefore in step 4, the calculated start time for the project, ST_i is set equal to PS_i (CST, True) = 6. From this calculated start time, the project finish time can then be determined, taking into account the other scheduling information.

[0042] In the foregoing example, the proposed new starting time CST is adjusted in Step 3 by adding the value of FT_i (CST, True) - [FT_i (CST, False) x FE_i], or δ (whichever is greater), to the prior proposed starting time, i.e. the project is moved to a later time. In some situations, it may be preferable to move the proposed starting time to an earlier point, i.e. to decrement the prior value of CST. For instance, this approach may be used to determine when a project needs to be completed on the basis of the latest projected finish time for projects that are already scheduled. It can be used to determine when work on a project must start in order to complete an entire program that is composed of several individual projects.

10050.0214551850

[0043] Along these same lines, it is also feasible to employ the principles of the invention to a reverse process flow, by utilizing a latest finish time in place of $EPST_i$, and calculate an appropriate starting time which ensures that the project will be finished by that time.

[0044] Other variants of the invention are also possible. For instance, to increase the speed of calculation, the value for flow time without resource contention, FT_i(CST, False), can be approximated from the earliest permissible start time by determining the value for $FT_i(EPST_i$, False), rather than recalculate it each time. For similar reasons, it may be preferable to set ST_i equal to CST, rather than PS_i(CST, True), in Step 4 of Figure 5.

[0045] In the foregoing example, the calculated start time ST_i is represented in terms of an absolute value, e.g. time 0, time 6, etc. As an alternative, it can be represented as an offset from the start time of one or more other projects, e.g. $ST_i = ST_{i-1} + 5$.

[0046] One embodiment of this invention can be embedded in a computerized project scheduling system that allows management, synchronization, tracking and what-if analyses for multiple projects. Single project scheduling can be part of such a system. In addition, the present invention can form a core piece of the system, and used to determine how projects will be paced over time.

[0047] It will be appreciated by those of ordinary skill in the art, that the present invention can be embodied in other forms without departing from the spirit or essential characteristics thereof. The foregoing description is considered illustrative and not restrictive. The scope of the invention is indicated by the following claims, and all changes that come within the meaning and range of equivalents are therefore intended to be embraced therein.

09851425.050201